

Analisis Code Review Menggunakan SonarQube Terhadap Aplikasi Rumah Kreatif Toba Berbasis Website

Nehemia Sitorus^{1*}, Asri Sirait², Efran Lumbantoruan³, Frans Panjaitan⁴, Handika Sukri Husni Harahap⁵, Christian Yohanes Simangunsong⁶

^{1,2,3,4,5,6} Fakultas Vokasi, Institut Teknologi Del, Indonesia

*strsnehemia88@gmail.com

<https://doi.org/10.56427/jcbd.v4i1.606>

INFO ARTIKEL

Histori Artikel

Diterima: 15 November 2024

Direvisi: 13 January 2025

Disetujui: 16 January 2025

Kata Kunci

SonarQube

Analisis Kode

Kualitas Perangkat Lunak

Keamanan Aplikasi

Platform Digital UMKM

Keywords

SonarQube

Code Analysis

Software Quality

Application Security

MSMEs Digital Platform

ABSTRAK

Rumah Kreatif Toba merupakan platform digital yang diakses melalui situs resmi <https://kreatif.tobakab.go.id> dan dirancang untuk mendukung pengelolaan serta pemasaran produk Usaha Mikro, Kecil, dan Menengah (UMKM) di Kabupaten Toba. Penelitian ini bertujuan untuk mengevaluasi efektivitas penerapan tool-assisted review menggunakan SonarQube dalam pengembangan aplikasi tersebut. Metode yang digunakan meliputi analisis kode sumber aplikasi dengan SonarQube, yang mendeteksi 36 masalah kritis (blocker), 227 titik kerentanan keamanan, dan tingkat duplikasi kode sebesar 38,8%. Hasil penelitian menunjukkan bahwa penggunaan SonarQube efektif dalam mengidentifikasi area yang perlu diperbaiki, terutama terkait keamanan, efisiensi, dan kemudahan pemeliharaan kode. Kesimpulannya, alat analisis kode seperti SonarQube sangat penting untuk meningkatkan kualitas dan keamanan perangkat lunak. Penelitian ini merekomendasikan agar penelitian di masa depan mempertimbangkan penggunaan alat dan metode analisis lain untuk memperoleh wawasan yang lebih komprehensif.

Rumah Kreatif Toba is a digital platform accessible through the official website <https://kreatif.tobakab.go.id> and is designed to support the management and marketing of Micro, Small, and Medium Enterprises (MSMEs) products in Toba Regency. This study aims to evaluate the effectiveness of tool-assisted reviews using SonarQube in the development of the application. The method used involves analyzing the application's source code with SonarQube, which detected 36 critical issues (blockers), 227 security vulnerabilities, and a code duplication rate of 38.8%. The results show that the use of SonarQube is effective in identifying areas that need improvement, particularly in terms of security, efficiency, and code maintainability. In conclusion, code analysis tools like SonarQube are essential for improving software quality and security. This study recommends that future research consider using other tools and analytical methods to gain more comprehensive insights.



JCBD is licensed under a [Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License](https://creativecommons.org/licenses/by-nc-sa/4.0/).

1. Pendahuluan

Dalam pengembangan perangkat lunak modern, kualitas kode yang andal dan bebas dari kesalahan merupakan aspek yang sangat penting. Salah satu metode untuk memastikan kualitas kode adalah melalui proses code review. Code review adalah proses peninjauan kode yang dilakukan oleh pengembang lain sebelum kode tersebut diintegrasikan ke dalam basis kode utama. Proses ini bertujuan untuk mendeteksi kesalahan, memastikan kepatuhan terhadap standar pengkodean, serta meningkatkan kualitas kode secara keseluruhan [1]. Untuk meningkatkan efektivitas proses ini, teknik tool-assisted review semakin banyak digunakan [2]. Teknik ini memanfaatkan alat bantu otomatis untuk mendeteksi kesalahan kode dan memberikan rekomendasi perbaikan secara lebih efisien [3].

Penggunaan tool-assisted review menawarkan berbagai keunggulan jika dibandingkan dengan code review manual. Alat bantu otomatis memungkinkan pengembang untuk mendeteksi kesalahan lebih cepat, mengidentifikasi kerentanan keamanan, serta mengurangi risiko pengulangan kesalahan[4]. Sebagai contoh, penelitian yang dilakukan oleh Google menunjukkan bahwa penggunaan alat bantu dalam code review dapat mempercepat proses pengembangan perangkat lunak dan meningkatkan kolaborasi tim[5]. Salah satu alat bantu yang telah banyak digunakan dalam proses ini adalah SonarQube. SonarQube merupakan alat analisis kualitas kode open-source yang mampu mengidentifikasi kesalahan sintaksis, duplikasi kode, code smell, serta kerentanan keamanan. Dengan menggunakan SonarQube, pengembang dapat meningkatkan keandalan, keamanan, dan efisiensi pengelolaan kode pada pengembangan perangkat lunak berbasis web[6].

Aplikasi Rumah Kreatif Toba berbasis website merupakan platform digital yang bertujuan untuk mendukung pengelolaan usaha mikro, kecil, dan menengah (UMKM) lokal di wilayah Toba[7]. Aplikasi ini memfasilitasi pemasaran produk UMKM secara online, memungkinkan bisnis untuk memperluas jangkauan pasar dan meningkatkan visibilitas produk mereka. Aplikasi Rumah Kreatif Toba adalah platform digital yang dapat diakses melalui website resmi di <https://kreatif.tobakab.go.id/>, yang bertujuan untuk mendukung pengelolaan dan pemasaran produk Usaha Mikro, Kecil, dan Menengah (UMKM) di Kabupaten Toba. Sistem ini beroperasi dengan memungkinkan pengguna untuk mendaftar dan mempromosikan produk mereka, sambil menyediakan antarmuka yang ramah pengguna bagi pelanggan. Pengelolaan kode aplikasi ini membutuhkan kontrol kualitas yang ketat untuk memastikan bahwa sistem berfungsi secara efektif, aman, dan dengan kesalahan yang minimal. Oleh karena itu, penerapan tool-assisted review menggunakan SonarQube dalam pengembangan aplikasi Rumah Kreatif Toba diharapkan dapat meningkatkan efektivitas proses review kode, mempercepat pengembangan fitur baru, dan mengurangi kesalahan yang dapat memengaruhi operasional aplikasi.

Penelitian ini berfokus pada analisis efektivitas penerapan proses code review menggunakan SonarQube dalam pengembangan aplikasi Rumah Kreatif Toba berbasis website. Dalam konteks aplikasi ini, fitur-fitur seperti manajemen produk, pemesanan, dan pembayaran memerlukan kode yang berkualitas tinggi untuk memastikan kinerja dan keamanan yang optimal. Dengan mengukur waktu yang diperlukan untuk menyelesaikan review, jumlah kesalahan yang terdeteksi, dan tingkat kepuasan pengembang, penelitian ini bertujuan memberikan wawasan mendalam tentang bagaimana SonarQube dapat mengidentifikasi masalah spesifik, seperti kerentanan dalam sistem pembayaran atau duplikasi kode yang dapat menghambat pengembangan fitur baru. Hasil dari penelitian ini diharapkan dapat memberikan gambaran yang lebih jelas mengenai keunggulan dan tantangan dari penggunaan SonarQube dalam pengembangan aplikasi berbasis web. Penelitian sebelumnya telah menunjukkan bahwa tool-assisted review dapat mengurangi beban kerja pengembang dan meningkatkan produktivitas tim secara keseluruhan [6]. Hasil penelitian diharapkan dapat mengungkap keunggulan dan tantangan yang dihadapi aplikasi Rumah Kreatif Toba, seperti kompleksitas integrasi dengan sistem pihak ketiga dan kebutuhan untuk memenuhi standar keamanan data pengguna. Dengan demikian, rekomendasi praktis yang dihasilkan dari penelitian ini akan membantu pengembang dan manajer proyek dalam mengoptimalkan proses code review, sehingga meningkatkan kualitas dan keberlanjutan aplikasi. Penelitian ini tidak hanya menyoroti manfaat penggunaan tool-assisted review, tetapi juga relevansinya dalam mendukung pengembangan aplikasi yang berfokus pada UMKM, memberikan kontribusi signifikan terhadap peningkatan kualitas dan keandalan aplikasi tersebut.

Code review merupakan salah satu langkah penting dalam memastikan kualitas perangkat lunak. Dalam penelitian oleh Ivimey-Cook dkk. (2023), code review diakui sebagai metode efektif untuk meningkatkan keandalan dan keterulangan hasil penelitian yang menggunakan pemrograman berbasis kode. Proses ini mengurangi kesalahan konseptual, programatik, dan sintaksis yang sering muncul selama pengembangan perangkat lunak, terutama dalam bidang sains dan teknologi[8].

Fadhli dkk. (2024) mengungkapkan bahwa deteksi code smells sangat penting untuk menjaga kualitas perangkat lunak agar mudah dikembangkan dan dimodifikasi di masa depan. Tools seperti SonarQube telah terbukti mampu mendeteksi berbagai code smells seperti large class, long method, dan duplicate code. Penelitian ini menyoroti pentingnya inspeksi kode untuk menjaga keberlanjutan dan kualitas perangkat lunak, terutama dalam aplikasi berbasis web[9].

Penelitian terkait menunjukkan berbagai alat bantu yang dapat digunakan untuk mendeteksi masalah pada kode, seperti SonarQube, PMD, dan CheckStyle. Ivimey-Cook dkk. (2023) menyebutkan bahwa lingkungan terintegrasi seperti Jupyter Notebook atau IDE dengan dukungan untuk pipeline data dapat membantu meningkatkan keterbacaan dan reusabilitas kode dalam proyek-proyek ilmiah. Ivimey-Cook dkk. (2023) dan Rahmawati dan Susetyo (2023) sepakat bahwa salah satu tantangan terbesar dalam code review adalah manajemen kompleksitas kode. Untuk mengatasinya, kedua penelitian menyarankan penggunaan alat bantu otomatis seperti SonarQube dan pengadopsian budaya kolaborasi dalam tim pengembang. Struktur proyek

yang rapi dan metadata yang lengkap juga dianggap penting untuk mendukung proses review yang lebih efisien [8].

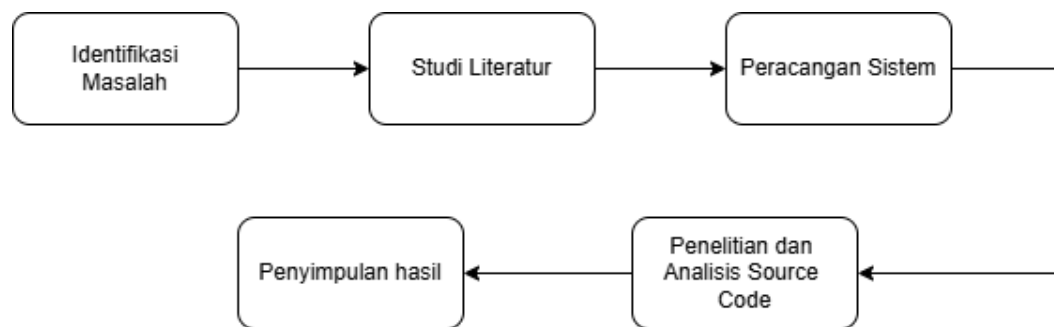
Studi oleh Fadhli dkk. (2024) mengenai aplikasi Bank Sampah menunjukkan bahwa implementasi deteksi code smells menggunakan SonarQube berhasil mengidentifikasi dan mengurangi masalah dalam desain perangkat lunak. Penelitian ini menjadi contoh bagaimana integrasi alat bantu dalam siklus pengembangan perangkat lunak dapat meningkatkan kualitas kode dan efisiensi pengelolaan proyek[9].

Pada penelitian Jurnal Improving Automated Code Reviews: Learning from Experience ini menyoroti implementasi code review di Google yang memadukan antara manual review dan penggunaan alat otomatis untuk meningkatkan efisiensi dan konsistensi kualitas kode. Google menggunakan alat seperti Critique yang mendeteksi kesalahan sintaksis dan masalah struktural lainnya secara otomatis. Namun, meskipun alat ini membantu meningkatkan kecepatan, manual review tetap diperlukan untuk menilai aspek-aspek logis dan desain dari kode yang tidak bisa ditangani secara efektif oleh alat otomatis. Temuan ini relevan dengan pengembangan website Humata Coffee dan Pizza Andaliman, karena mereka bisa memanfaatkan kombinasi alat otomatis dengan peninjauan manual untuk mendapatkan hasil terbaik dalam kualitas kode IEEE XP.

Jurnal Modern Code Review: A Case Study at Google, mengeksplorasi penerapan modern code review di Google, di mana pendekatan manual dan otomatis digunakan secara bersamaan. Temuan penelitian menunjukkan bahwa alat bantu seperti Critique membantu dalam mendeteksi kesalahan sintaksis dan meningkatkan efisiensi tim. Namun, proses manual masih diperlukan untuk menangkap kesalahan konseptual yang kompleks dan memastikan kualitas logika bisnis. Studi ini relevan untuk memahami keseimbangan antara metode manual dan otomatis dalam proyek berskala besar.

2. Metodologi Penelitian

Penelitian ini berfokus pada analisis code dalam pengembangan website Rumah Kreatif Toba (RKT). Metode penelitian mencakup beberapa tahap, yaitu Identifikasi Masalah, Studi Literatur, Perancangan Sistem, Penelitian Analisa Source Code dan Penyimpulan hasil. Berikut adalah tahapan metode penelitian ini [6]:



Gambar 1. Metode Penelitian

Penelitian ini dimulai dengan identifikasi masalah dalam pengembangan kode website Rumah Kreatif Toba, di mana masalah utamanya adalah sulitnya mendeteksi kesalahan kode (defects) secara cepat dan efisien dalam proses review kode manual. Hal ini berdampak pada menurunnya kualitas kode dan memperlambat waktu pengembangan. Untuk mendukung penelitian, dilakukan studi literatur dengan mengumpulkan referensi terkait teknik code review, penggunaan alat bantu otomatis seperti SonarQube, serta studi kasus aplikasi open source berbasis Laravel dan Vue.js. Studi ini bertujuan untuk mendapatkan dasar teori yang relevan dan mendukung proses analisis kualitas kode.

Pada tahap perancangan sistem, dilakukan berbagai langkah teknis untuk memastikan kelancaran implementasi. Peneliti menyiapkan infrastruktur SonarQube pada virtual machine berbasis sistem operasi Linux, dengan mengonfigurasi kebutuhan seperti OpenJDK 11 atau JRE 11 sebagai pendukung platform dan PostgreSQL versi 13 untuk menyimpan data hasil analisis kode. Selain itu, disusun alur kerja review kode yang melibatkan pengunggahan kode ke SonarQube, analisis otomatis, diskusi hasil analisis oleh tim, serta implementasi perbaikan kode. Peran dalam tim juga ditentukan, yaitu Author sebagai penulis kode, Moderator sebagai pemimpin proses review, dan Reviewer sebagai peninjau kode.

Tahap pengujian dan analisis kode dilakukan dengan menggunakan SonarQube, diawali dengan proses analisis awal untuk mendeteksi bugs, code smells, vulnerabilities, dan duplikasi kode. Hasil analisis ditampilkan dalam dashboard SonarQube dengan indikator Quality Gate untuk mengevaluasi kelayakan kode. Temuan hasil analisis kemudian didiskusikan oleh Reviewer, yang memberikan rekomendasi perbaikan

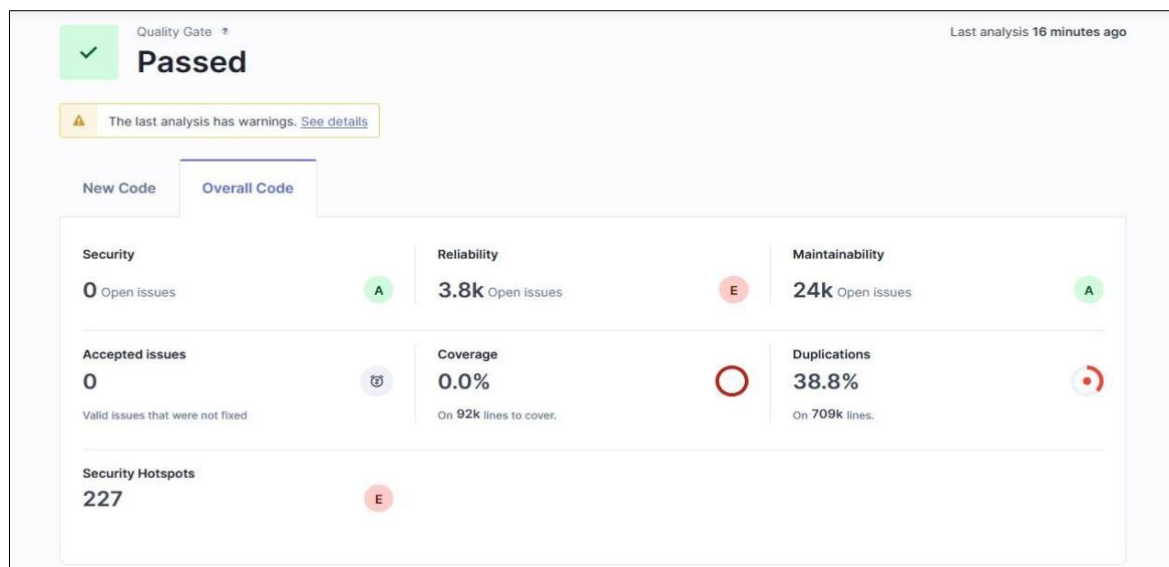
kepada Author. Author melakukan perbaikan kode berdasarkan rekomendasi tersebut, dan setelah itu dilakukan analisis ulang untuk memastikan kualitas kode telah meningkat sesuai standar yang ditetapkan.

Terakhir, dilakukan penyimpulan hasil dengan membandingkan hasil analisis terhadap indikator keberhasilan, yaitu jumlah kesalahan yang ditemukan dan diperbaiki, peningkatan kualitas kode berdasarkan skor code maintainability, serta efisiensi waktu yang dihabiskan dalam setiap tahap proses review. Hasil penelitian ini digunakan untuk mengevaluasi efektivitas penerapan tool-assisted review berbasis SonarQube dalam meningkatkan kualitas kode aplikasi berbasis web. SonarQube adalah platform open-source yang dikembangkan oleh SonarSource, digunakan untuk inspeksi berkelanjutan pada kualitas kode aplikasi. Lebih dari itu, SonarQube juga dapat melakukan analisis kode secara statis, yang mana menyediakan laporan lebih detail terkait bug, code smells, vulnerability hingga duplication code. SonarQube didukung dengan 30+ bahasa pemrograman yang terbagi ke dalam beberapa versi, yaitu 17 bahasa pemrograman untuk Community Edition, 24 bahasa pemrograman Developer Edition, 29 bahasa pemrograman untuk Enterprise dan Data Center Edition. SonarQube mengklasifikasikan aturan ke dalam 5 tingkatan, yaitu Blocker, Critical, Major, Minor, dan Info. Setiap aturan pengkodean dalam SonarQube mendefinisikan sebuah Quality Gate. Quality Gate akan menentukan status Passed atau Failed sebuah source code dan menampilkan penilaian untuk masalah yang ditemukan

3. Hasil dan Pembahasan

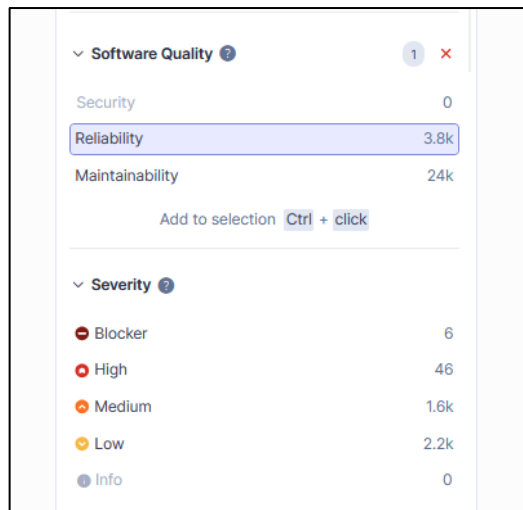
Penelitian ini menganalisis kode sumber aplikasi Rumah Kreatif Toba, sebuah platform digital yang mendukung pengelolaan dan pemasaran produk Usaha Mikro, Kecil, dan Menengah (UMKM) di Kabupaten Toba. Platform ini memainkan peran penting dalam mempromosikan produk lokal secara online, sehingga kualitas perangkat lunaknya perlu dijaga untuk mendukung stabilitas layanan kepada pengguna. Kode sumber aplikasi ini dianalisis menggunakan alat bantu SonarQube, yang memberikan laporan komprehensif terkait masalah seperti blocker issues, critical issues, vulnerabilities, code smells, dan duplicated code.

Proses analisis dilakukan dengan mengelompokkan temuan berdasarkan tingkat keparahan (severity) dan jenis masalah yang diidentifikasi. Sebagai contoh, ditemukan 36 blocker issues dan 227 vulnerabilities, yang dapat mempengaruhi integrity serta efficiency aplikasi. Hasil dari analisis ini memberikan wawasan penting tentang langkah-langkah perbaikan yang perlu dilakukan, seperti mengurangi duplikasi kode sebesar 38,8% untuk meningkatkan efficiency dan memperbaiki kerentanan keamanan untuk meningkatkan integrity aplikasi. Gambar 2 merangkum temuan utama yang menjadi fokus analisis. Metode penelitian dimulai dengan mengidentifikasi masalah terkait kualitas dan keamanan aplikasi Rumah Kreatif Toba, diikuti dengan studi literatur mengenai alat analisis kode, khususnya SonarQube. Setelah mendeskripsikan aplikasi, kode sumber dikumpulkan dan dianalisis menggunakan SonarQube, yang mengungkapkan berbagai metrik seperti 3.8k masalah terbuka, 24k isu maintainability, 0.0% coverage, 38.8% duplikasi, dan 227 titik panas keamanan. Hasil analisis digunakan untuk mengidentifikasi area perbaikan dan merekomendasikan peningkatan pengujian unit, pengurangan duplikasi kode, serta penanganan masalah keamanan. Perbaikan diimplementasikan dan dievaluasi untuk menilai efektivitas SonarQube dalam meningkatkan kualitas aplikasi.



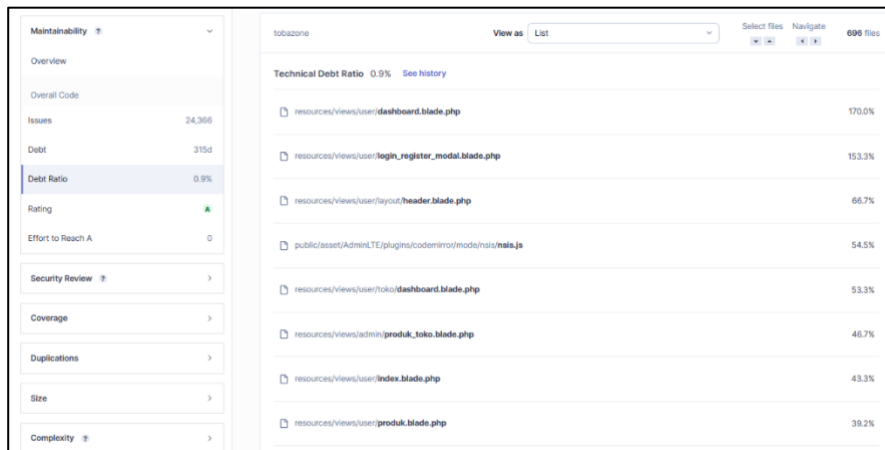
Gambar 2. Report Analisis Project Rumah Kreatif Toba

Laporan hasil analisis dari proses scanning SonarQube merupakan temuan keseluruhan kode dari pengujian pada project Rumah Kreatif Toba. Menurut hasil pada Gambar 2, status Quality Gate pada Overall Code dinyatakan Passed dan berwarna hijau. Hal ini disebabkan penilaian pada hasil analisis di berbagai matriks yang telah dijalankan oleh SonarScanner. Hasil dari analisis itu menampilkan bahwa tidak terdapat open issue pada bagian Security yang ditemukan pada code base, sehingga mendapatkan status A pada securitynya. Terdapat 3,8k open issue pada bagian Reliability yang ditemukan pada code base, sehingga mendapatkan status E. Status peringkat Reliability adalah E jika terdapat setidaknya satu masalah Reliability dengan tingkat keparahan issue yaitu Blocker. Tingkat keparahan pada Reliability dapat dilihat pada Gambar 3, yaitu terdapat 36 issue pada tingkat Blocker, 48 issue pada tingkat High, 1,6k issue pada tingkat Medium dan 2,2k issue pada tingkat Low.



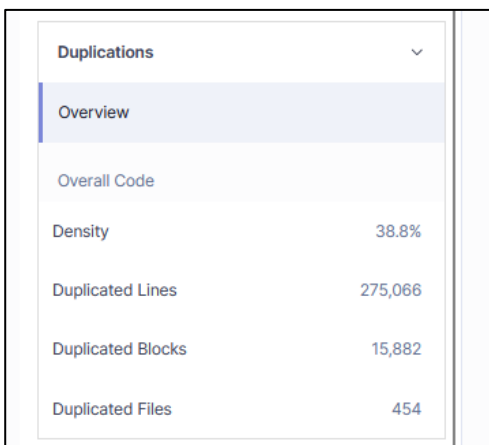
Gambar 3. Severity pada masalah Reliability

Terdapat 24k open issue pada bagian Maintainability yang ditemukan, tetapi menampilkan status A pada Maintainability. Peringkat Maintainability adalah A jika Debt ratio kurang dari 5,0%. Pada Gambar 4, ditampilkan bahwa Debt Ratio yang dimiliki oleh kode yaitu 0,9% sehingga masih termasuk kedalam kategori A.



Gambar 4. Debt Ratio Maintainability

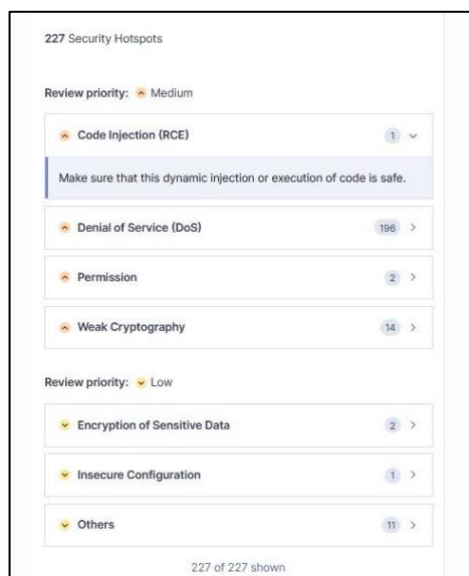
Terdapat 38,8% Duplication yang ditemukan pada code base, dengan informasi yang ditampilkan pada Gambar 5. Dengan informasi bahwa terdapat 38,8% Density, 275.066 Duplicated Lines, 15.882 Duplicated Blocks dan 454 Duplicate Files.



Gambar 5. Informasi Duplications pada Overall Code

Gambar 6 selanjutnya menunjukkan detail analisis "Security Hotspots" yang ditemukan dalam proyek ini. Terdapat total 227 Security Hotspots dengan prioritas tinjauan yang terbagi menjadi dua tingkatan, yaitu Medium dan Low. Pada prioritas Medium, masalah keamanan utama yang ditemukan adalah Code Injection (RCE), yang membutuhkan perhatian segera untuk memastikan tidak ada injeksi atau eksekusi kode yang berbahaya. Selain itu, terdapat 196 hotspot terkait potensi Denial of Service (DoS), yang dapat mengganggu ketersediaan layanan. Masalah lain yang teridentifikasi meliputi Permission dan Weak Cryptography, yang memerlukan tinjauan lebih lanjut untuk memastikan akses yang aman dan penggunaan algoritma enkripsi yang kuat.

Pada prioritas Low, Encryption of Sensitive Data dan Insecure Configuration menjadi dua isu yang termasuk dalam kategori risiko rendah. Namun demikian, perbaikan tetap disarankan untuk mencegah eskalasi risiko di masa mendatang. Kategori Others mencakup 11 hotspot, yang kemungkinan mencerminkan masalah keamanan lainnya yang lebih bervariasi.



Gambar 6. Rincian Analisis Security Hotspots

Tabel 1 berikut memberikan gambaran rinci mengenai distribusi Lines of Code (LoC), metrik keamanan, keandalan, kemudahan pemeliharaan, hotspot keamanan, cakupan pengujian (coverage), dan tingkat duplikasi kode di setiap direktori proyek "Rumah Kreatif Toba". Dari hasil analisis, ditemukan bahwa directory public memiliki jumlah LoC yang sangat tinggi, yaitu sebesar 564.139 baris kode. Angka ini menunjukkan bahwa sebagian besar aktivitas aplikasi terkait dengan elemen publik, seperti file statis atau aset. Selain itu, directory ini mencatat nilai Maintainability sebesar 23.594, yang mengindikasikan potensi adanya kode yang sulit dirawat dan membutuhkan perhatian lebih. Tingkat duplikasi pada directory ini mencapai 39,6%, yang

merupakan angka tertinggi dibandingkan directory lain. Hal ini dapat menyebabkan redundansi dan memperbesar risiko inkonsistensi kode. Directory public juga mencatat jumlah Security Hotspots tertinggi, yaitu sebanyak 219, menjadikannya area dengan prioritas tinggi untuk ditinjau lebih lanjut.

Data dalam gambar diperoleh dari analisis kode aplikasi Rumah Kreatif Toba menggunakan SonarQube, dimulai dengan pengumpulan kode sumber, instalasi dan konfigurasi alat, lalu menjalankan analisis untuk mendeteksi metrik seperti baris kode, masalah keamanan, keandalan, maintainability, titik panas keamanan, coverage, dan duplikasi. Hasilnya disusun dalam tabel untuk menggambarkan kualitas dan keamanan setiap direktori aplikasi.

Tabel 1 Informasi Setiap Directory dalam Rumah Kreatif Toba

Project Name	Lines of Code	Security	Reliability	Maintainability	Security Hotspot	Coverage	Duplications
app	2.607	0	3	162	1	0.0%	19.3%
bootstrap	16	0	0	0	0	0.0%	0.0%
config	552	0	0	2	1	0.0%	0.0%
database	608	0	0	24	0	0.0%	0.0%
public	564.139	0	3.627	23.594	219	0.0%	39.6%
resources	725	0	197	572	6	0.0%	14.0%
routes	112	0	0	9	0	0.0%	45.9%
tests	37	0	1	1	0	0.0%	0.0%
phpunit.xml	29	0	0	1	0	-	0.0%
server.php	7	0	0	1	0	0.0%	0.0%

Directory app, dengan LoC sebesar 2.607, memiliki tiga masalah keandalan dan satu Security Hotspot. Tingkat Maintainability-nya mencapai 162, yang relatif rendah dibandingkan directory public, sedangkan tingkat duplikasi kode mencapai 19,3%. Directory routes, meskipun hanya memiliki LoC sebesar 112, mencatat tingkat duplikasi yang sangat tinggi, yaitu 45,9%. Ini menunjukkan bahwa meskipun ukuran directory kecil, tetap diperlukan upaya untuk meningkatkan kualitas kode. Directory resources memiliki enam Security Hotspots dengan tingkat duplikasi sebesar 14%. Tingkat Maintainability yang cukup tinggi, yaitu 572, menunjukkan bahwa meskipun terdapat potensi kerumitan kode, directory ini memiliki kualitas yang lebih baik dibandingkan directory lain. Sementara itu, directory config memiliki satu Security Hotspot dengan Maintainability rendah sebesar 2 poin, dan database mencatat 24 poin Maintainability tanpa adanya Security Hotspots.

4. Kesimpulan

Dalam era digital yang semakin berkembang, penelitian ini memberikan kontribusi penting terhadap pemahaman kualitas kode dalam pengembangan perangkat lunak. Hasil penelitian menunjukkan bahwa penerapan SonarQube dalam analisis kode sumber aplikasi Rumah Kreatif Toba efektif dalam mengidentifikasi berbagai masalah yang perlu diperbaiki. Secara khusus, SonarQube berhasil mendeteksi 36 masalah kritis (blocker), 227 titik kerentanan keamanan, dan adanya duplikasi kode sebesar 38,8%. Temuan ini menggambarkan pentingnya penggunaan alat analisis statis untuk meningkatkan kualitas dan keamanan perangkat lunak, serta memberikan wawasan berharga bagi pengembang untuk melakukan perbaikan yang diperlukan dalam kode mereka. Penggunaan SonarQube tidak hanya membantu dalam mendeteksi masalah, tetapi juga berkontribusi pada proses pengembangan yang lebih efisien dan berkelanjutan.

Bagi peneliti yang ingin melanjutkan penelitian ini, disarankan untuk mempertimbangkan penggunaan alat analisis kode lainnya atau metode yang berbeda, seperti analisis statis dan dinamis, untuk memberikan wawasan yang lebih komprehensif tentang kualitas perangkat lunak yang sedang dikembangkan.

Ucapan Terima Kasih

Kami mengucapkan terima kasih yang sebesar-besarnya kepada semua pihak yang telah berkontribusi dalam penelitian ini. Pertama, kepada tim pengembang Rumah Kreatif Toba yang telah memberikan dukungan dan kerjasama dalam proses pengembangan aplikasi. Terima kasih juga kepada Fakultas Vokasi, Institut Teknologi Del, yang telah menyediakan fasilitas dan sumber daya yang diperlukan untuk penelitian ini. Kami juga mengucapkan terima kasih kepada para penelaah dan rekan-rekan yang telah memberikan masukan berharga selama proses review, serta kepada semua mahasiswa yang turut berpartisipasi dalam pengumpulan data dan analisis. Akhirnya, terima kasih kepada keluarga dan teman-teman yang selalu memberikan dukungan

moral. Semoga hasil penelitian ini dapat bermanfaat bagi pengembangan perangkat lunak dan meningkatkan kualitas aplikasi berbasis web di masa mendatang.

Referensi

- [1] C. Sadowski, E. Söderberg, L. Church, M. Sipko, and A. Bacchelli, “Modern code review: A case study at google,” *Proc. - Int. Conf. Softw. Eng.*, pp. 181–190, 2018, doi: 10.1145/3183519.3183525.
- [2] Y. Kamei *et al.*, “A large-scale empirical study of just-in-time quality assurance,” *IEEE Trans. Softw. Eng.*, vol. 39, no. 6, pp. 757–773, 2013, doi: 10.1109/TSE.2012.70.
- [3] S. McIntosh, Y. Kamei, B. Adams, and A. E. Hassan, “An empirical study of the impact of modern code review practices on software quality,” *Empir. Softw. Eng.*, vol. 21, no. 5, pp. 2146–2189, 2016, doi: 10.1007/s10664-015-9381-9.
- [4] P. W. Gonçalves, E. Fregnan, T. Baum, K. Schneider, and A. Bacchelli, “Do Explicit Review Strategies Improve Code Review Performance?,” *Proc. - 2020 IEEE/ACM 17th Int. Conf. Min. Softw. Repos. MSR 2020*, pp. 606–610, 2020, doi: 10.1145/3379597.3387509.
- [5] H. Y. Lin, P. Thongtanunam, C. Treude, and W. Charoenwet, “Improving Automated Code Reviews: Learning from Experience,” *Proc. - 2024 IEEE/ACM 21st Int. Conf. Min. Softw. Repos. MSR 2024*, pp. 278–283, 2024, doi: 10.1145/3643991.3644910.
- [6] A. F. Rahmawati and Y. A. Susetyo, “Analisis Quality Code Menggunakan Sonarqube Dalam Suatu Aplikasi Berbasis Laravel,” *IT-Explore J. Penerapan Teknol. Inf. dan Komun.*, vol. 2, no. 2, pp. 99–103, 2023, doi: 10.24246/itexplore.v2i2.2023.pp99-103.
- [7] Z. Sufiani, “Definisi, Kriteria dan Konsep UMKM,” *Osfpreprints*, no. 90500120021, pp. 1–13, 2022.
- [8] E. R. Ivimey-Cook *et al.*, “Implementing code review in the scientific workflow: Insights from ecology and evolutionary biology,” *J. Evol. Biol.*, vol. 36, no. 10, pp. 1347–1356, 2023, doi: 10.1111/jeb.14230.
- [9] C. Smell and A. B. Sampah, “Main Program A genda,” *Proc. ASIL Annu. Meet.*, vol. 112, pp. 351–382, 2018, doi: 10.1017/amp.2019.127.
- [10] L. R. Ilmi, R. Prahesti, P. Pratana, A. Wahyuningsih, and C. Manuela, “Review Ketepatan Kode Diagnosis Dan Pending Klaim Rekam Medis Pasien Covid-19 Tahun 2021,” *J. Ilm. Perkam dan Inf. Kesehat. Imelda*, vol. 8, no. 1, pp. 9–16, 2023, doi: 10.52943/jipiki.v8i1.1141.
- [11] S. Wahyuningsih, “Sri Wahyuningsih Peranan UKM Dalam Perekonomian Indonesia,” *Mediagro*, vol. 5, no. 1, pp. 1–14, 2009.
- [12] I. sri wahyuningsi Manguling and J. M. Parenreng, “Security System Analysis Using the HTTP Protocol Against Packet Sniffing Attacks,” *Internet Things Artif. Intell. J.*, vol. 3, no. 4, pp. 325–340, 2023, doi: 10.31763/iota.v3i4.612.
- [13] Lidya Adriani Darma, Rini Sovia, and Heriyanto, “Penerapan E-Commerce Model Business To Customer D’Sruput Cabang Lubuk Begalung Berbasis Web,” *J. Comput. Digit. Bus.*, vol. 1, no. 2, pp. 65–71, 2022, doi: 10.56427/jcbd.v1i2.14.
- [14] P. D. Marketing and S. L. Review, “Perkembangan Digital Marketing Systematic Literatur Review,” vol. 3, no. 3, pp. 105–111, 2024.